Amendments to the Claims

This listing of claims replaces all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Currently Amended) A method for extracting a channel from a data stream~~, said method consisting of~~ using a modified fast convolution algorithm, said modified fast convolution algorithm consisting of a common-channel part ~~common to all channels~~ followed by a channel-specific part, said channel-specific part ~~characterized by~~ comprising the steps of:

selecting a range of n Discrete Fourier Transform bins around the center frequency of the channel;

multiplying said bins with a frequency response; [and]

performing an N_{IDFT}-point Inverse Discrete Fourier Transform on these n data points; and,

performing a signal processing step.

2. (Currently Amended) The method of claim 1, ~~further characterized~~ wherein[[:]]

said common-channel part of said modified fast convolution algorithm [[has a]] comprises the step of performing a N_{FFT}-Point Fast Fourier Transform on overlapping blocks of said data stream.

3. (Currently Amended) The method of claim 2, ~~further characterized~~ wherein[[:]]

said N_{FFT}-point Fast Fourier Transform in said common-channel part of said modified fast convolution algorithm is preceded by the steps of:

first, processing said data stream by a η% overlap block generator: [[and]]

second, multiplexing said data stream to form a complex signal;

[[while]] wherein said channel-specific part of said modified fast convolution ~~algorithm has~~ algorithm further comprises the steps of:

a first step of performing extraction of said bins;

a second step of performing said multiplication of said bins with said frequency response;

a third step of performing an N_{IDFT}-point Inverse Discrete Fourier Transform on these n data points; and,

a fourth step of processing said digital data stream by a η% overlap block combiner.

4. (Currently Amended) The method of claim 1, ~~further characterized~~ wherein said frequency response has a limited range.

5. (Currently Amended) The method of claim 3, wherein said η% overlap block generator ~~is further characterized wherein:~~

generates said blocks ~~are generated~~ using an overlap/add process which chops said data stream into non-overlapping sections of length N_{FFT}* (1-η) and padded with N_{FFT}*η zeros to form a single block.

6. (Currently Amended) The method of claim 3, wherein said η% overlap block generator ~~is further characterized wherein:~~

generates said blocks ~~are generated~~ using an overlap/save process which chops said data stream into a series of blocks of length N_{FFT}, each of which has an overlap with the previous block in the series given by a length of N_{FFT}*η.

7. (Currently Amended) The method of claim 3, wherein said η% overlap block combiner ~~is further characterized wherein:~~

processes said data stream ~~is processed~~ using an overlap/add process wherein said blocks are overlapped with the previous block by a length equal to N_{IDFT}*η, the overlapping part of a block is added to the previous block's corresponding overlapping part to produce the output data stream.

8. (Currently Amended) The method of claim 3<u>,</u> wherein η% overlap block combiner ~~is further characterized wherein:~~

<u>processes</u> said data stream ~~is processed~~ using an overlap/save process wherein said blocks are overlapped with the previous block by a length equal to $N_{IDFT}*\eta$, the overlapping parts of the blocks are discarded <u>and</u> said output data stream ~~being~~ ~~form~~ <u>is formed</u> from the non-overlapping parts of the blocks.

9. (Currently Amended) The method of claim 3<u>,</u> wherein said multiplexing step [[is]] further ~~characterized by:~~ <u>comprises the step of</u>

producing a complex signal $z(t)=x(t)+j*y(t)$, where $x(t)$ and $y(t)$ are two consecutive blocks.

10. (Currently Amended) The method of claim 9<u>,</u> ~~further characterized~~ wherein[[:]]

~~said sequence~~ y(t) is ~~also~~ rotated.

11. (Currently Amended) The method of claim 3<u>,</u> ~~further characterized~~ wherein[[:]]

said N_{FFT}-point Fast Fourier Transform is a pipeline architecture with a power of 2 ~~and said bin extraction reorders the output from the Fast Fourier Transform and selects only the bins needed~~.

12. (Currently Amended) A method for inserting a channel into a data stream, said method consisting of a modified fast convolution algorithm, said modified fast convolution algorithm consisting of a channel-specific part followed by a common-channel part common to all channels, said channel-specific part ~~characterized by~~ <u>comprises the steps of</u>:

performing a signal processing step;

performing an N_{DFT}-point Discrete Fourier Transform on said stream;

multiplying said stream with a frequency response; and<u>,</u>

inserting a range of n Fast Fourier Transform bins around the center frequency of the channel.

13. (Currently Amended) The method of claim 12, ~~further characterized~~ wherein[[:]]

said common-channel part of said modified fast convolution algorithm ~~has a~~ comprises the step of performing a N_{IFFT}-point Inverse Fast Fourier Fast Transform on overlapping blocks of said data stream.

14. (Currently Amended) The ~~channelizer~~ method of claim 13, ~~further characterized~~ wherein[[:]]

said channel-specific part of said modified fast convolution ~~algorithm has~~ algorithm comprises the steps of:

a first step of processing said digital data stream by a η% overlap block generator;

~~followed by said~~ a second step of performing a Discrete Fourier Transform; ~~followed by~~

a third step of multiplying the result of said Discrete Fourier Transform with the filter frequency coefficients; and,

a fourth step of inserting said bins around the center frequency of the channel;

~~while~~ said common-channel part of said modified fast convolution ~~algorithm has~~ algorithm further comprises the steps of:

~~said step of performing am an N_{IFFT}-point Inverse Fast Fourier followed by a second step of~~ de-multiplexing the output from said N_{IFFT}-point Inverse Fast Fourier Transform to form a real signal; and,

~~a third step of~~ processing said digital data stream by a η% overlap block combiner.

15. (Currently Amended) The method of claim 12<u>,</u> ~~further characterized~~ wherein said frequency response has a limited range.

16. (Currently Amended) The method of claim 14<u>,</u> wherein said η% overlap block generator ~~is further characterized wherein:~~

<u>generates</u> said blocks ~~are generated~~ using an overlap/add process which chops said data stream into non-overlapping sections of length N_{FFT}*(1-η) and padded with N_{FFT}*η zeros to form a single block.

17. (Currently Amended) The method of claim 14<u>,</u> wherein said η% overlap block generator ~~is further characterized wherein:~~

<u>generates</u> said blocks ~~are generated~~ using an overlap/save process which chops said data stream into a series of blocks of length N_{FFT}, each of which has an overlap with the previous block in the series given by a length of N_{FFT}*η.

18. (Currently Amended) The method of claim 14<u>,</u> wherein said η% overlap block combiner ~~is further characterized wherein:~~

<u>processes</u> said data stream ~~is processed~~ using an overlap/add process wherein said blocks are overlapped with the previous block by a length equal to N_{IDFT}*η, the overlapping part of a block [[is]] <u>being</u> added to the previous block's corresponding overlapping part to produce the output data stream.

19. (Currently Amended) The method of claim 14<u>,</u> wherein η% overlap block combiner ~~is further characterized wherein:~~

<u>processes</u> said data stream ~~is process~~ using an overlap/save process wherein said blocks are overlapped with the previous block by a length equal to N_{IDFT}*η, the overlapping parts of the blocks are discarded <u>and</u> said output data stream ~~being form~~ <u>is formed</u> from the non-overlapping parts of the blocks.

20. (Currently Amended) The method of claim 14, ~~further characterized~~ wherein[[:]]

said bins are inserted into said Inverse Fast Fourier Transform in a symmetrical way where $Z(k_{start}+k)=X(k)$ and $Z(N_{IFFT}-k_{start}-k)=X'(k)$, K_{start} being where the first bin of the channel is to be inserted and K is an integer from $0 \rightarrow N-1$, said bins for a given channel given by $X(0) \rightarrow X(N-1)$ where $X'(k)$ is the complex conjugate of $X(k)$ and being inserted into said Inverse Fast Fourier Transform in the order $X(0) \rightarrow X(N-1)$.

21. (Currently Amended) The method of claim 14, ~~further characterized~~ wherein[[:]]

said bins are inserted into said Inverse Fast Fourier Transform by $Z(k_{start}+k)=X(k)+jY(k)$ and $Z(N_{IFFT}-k_{start}-k)=X'(k)+jY'(k)$, K_{start} being where the first bin of the channel is to be inserted and K is an integer from $0 \rightarrow N-1$, said bins for a given channel given by $X(O) \rightarrow X(N-1)$ where $X'(k)$ is the complex conjugate of $X(k)$ and being inserted into said Inverse Fast Fourier Transform in the order $X(0) \rightarrow X(N-1)$.

22-24. (Cancelled)

* * *